



# UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE  
United States Patent and Trademark Office  
Address: COMMISSIONER FOR PATENTS  
P.O. Box 1450  
Alexandria, Virginia 22313-1450  
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/083,784	02/26/2002	Brian Gerard Goodman	TUC920010094US1 (14942)	5475
7590 09/21/2005			EXAMINER	
STEVEN FISCHMAN, ESQ. SCULLY, SCOTT, MURPHY AND PRESSER 400 Garden City Plaza Garden City, NY 11530			RUTTEN, JAMES D	
			ART UNIT	PAPER NUMBER
			2192	

DATE MAILED: 09/21/2005

Please find below and/or attached an Office communication concerning this application or proceeding.

# Office Action Summary

Application No.

10/083,784

Applicant(s)

GOODMAN, BRIAN GERARD

Examiner

J. Derek Rutten

Art Unit

2192

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

## Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

## Status

- 1) ☒ Responsive to communication(s) filed on 08 July 2005.
- 2a) ☒ This action is **FINAL**. 2b) ☐ This action is non-final.
- 3) ☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

## Disposition of Claims

- 4) ☒ Claim(s) 1-44 is/are pending in the application.
- 4a) Of the above claim(s) \_\_\_\_\_ is/are withdrawn from consideration.
- 5) ☐ Claim(s) \_\_\_\_\_ is/are allowed.
- 6) ☒ Claim(s) 1-44 is/are rejected.
- 7) ☐ Claim(s) \_\_\_\_\_ is/are objected to.
- 8) ☐ Claim(s) \_\_\_\_\_ are subject to restriction and/or election requirement.

## Application Papers

- 9) ☐ The specification is objected to by the Examiner.
- 10) ☒ The drawing(s) filed on 29 April 2002 is/are: a) ☒ accepted or b) ☐ objected to by the Examiner.  
Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).  
Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).
- 11) ☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

## Priority under 35 U.S.C. § 119

- 12) ☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
- a) ☐ All b) ☐ Some \* c) ☐ None of:
- ☐ Certified copies of the priority documents have been received.
  - ☐ Certified copies of the priority documents have been received in Application No. \_\_\_\_\_.
  - ☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).
- \* See the attached detailed Office action for a list of the certified copies not received.

## Attachment(s)

- |  |   |
|--|---|
| 1) <input checked="" type="checkbox"/> Notice of References Cited (PTO-892)  | 4) <input type="checkbox"/> Interview Summary (PTO-413)<br>Paper No(s)/Mail Date. _____ |
| 2) <input type="checkbox"/> Notice of Draftsperson's Patent Drawing Review (PTO-948)                                   | 5) <input type="checkbox"/> Notice of Informal Patent Application (PTO-152)             |
| 3) <input type="checkbox"/> Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08)<br>Paper No(s)/Mail Date _____ | 6) <input type="checkbox"/> Other: _____  |

### **DETAILED ACTION**

1. Acknowledgement is made of Applicant's amendment dated 8 July 2005, responding to the 8 April 2005 Office action provided in the rejection of claims 1-44, wherein claims 1, 13, 17, 31, 32, and 44 have been amended, no claims have been canceled, and no new claims have been added. Claims 1-44 remain pending in the application and have been fully considered by the examiner.

2. Applicant's amendment necessitated the new ground(s) of rejection presented in this Office action. Accordingly, **THIS ACTION IS MADE FINAL**. See MPEP § 706.07(a). Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire THREE MONTHS from the mailing date of this action. In the event a first reply is filed within TWO MONTHS of the mailing date of this final action and the advisory action is not mailed until after the end of the THREE-MONTH shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the date of this final action.

### ***Response to Arguments***

3. Applicant's arguments on pages 17-19 of the reply, with respect to drawing, claim and specification objections, and 35 U.S.C. § 112 2<sup>nd</sup> paragraph rejections are convincing, and are therefor withdrawn.

Art Unit: 2192

4. Applicant's arguments, see pages 19-32, filed 8 July 2005, with respect to the rejection(s) of claim(s) 1, 6, 8-11, 13, 16, 18-21, 31, 37, and 39-42 under 35 U.S.C. § 102(a) have been fully considered and are persuasive. Therefore, the rejection has been withdrawn. However, upon further consideration, a new ground(s) of rejection is made in view of U.S. Patent No. 6,023,704 to Gerard et al.

*Claim Rejections - 35 USC § 112*

5. The following is a quotation of the first paragraph of 35 U.S.C. 112:

The specification shall contain a written description of the invention, and of the manner and process of making and using it, in such full, clear, concise, and exact terms as to enable any person skilled in the art to which it pertains, or with which it is most nearly connected, to make and use the same and shall set forth the best mode contemplated by the inventor of carrying out his invention.

6. Claims 1-44 are rejected under 35 U.S.C. 112, first paragraph, as failing to comply with the written description requirement. The claim(s) contains subject matter which was not described in the specification in such a way as to reasonably convey to one skilled in the relevant art that the inventor(s), at the time the application was filed, had possession of the claimed invention. Newly amended independent claims 1, 13, 31, 32, and 44 all contain limitations reciting "...and thereby prevent a task switching function that originates from the one or more code update routines of the incoming image from executing". Applicant has referred to page 14, lines 1-26 of the specification as support for this new claim language. Page 14 contains a description of the invention as it relates to Figure 3. In particular, lines 10-19 describes that task switching is executed when the code update routines of the incoming image retrieve the task switching offset from the "current" executing code image, and then transfer control to the current image using the offset. From page 14, lines 12-19 (emphasis added):

Art Unit: 2192

More particularly, in order to switch to a different process thread, a task switching function is to be utilized. Therefore, at step D the new code update routines of the executable code and data 310 in the incoming code image 303 retrieve the task switching routine offset 218 from the header area 206 of the executing code image 301. Thereafter at step E, microprocessor control is thus transferred to the task switching function in the executable code and data 210, which switches microprocessor control to any other process thread, i.e., effectively executing the other process thread from the executing code image 301. **Task switching is performed by a part of an operating system (not shown) of the embedded system 100, which is generally called a scheduler (not shown).**

While this passage clearly describes a task switching function that is executed by the “current” executing code image, the passage does not expressly describe a task switching function that originates from the “incoming” code update routines, and so cannot describe *preventing* a task switching function that originates from the code update routine. Figure 3 shows that the “incoming” code image contains a task switch routine, but the specification merely notes that it exists (see page 15 line 21) and does not describe any execution of this routine that is prevented. Further, the specification describes a “task switching function” that is to be performed (see page 14 line 11). In light of the totality of the cited passage, this appears to describe a task switching function that occurs while the “incoming” code update routines are executing. In this context, even if a task switch function originated from the “incoming” code update routines, and the task switching function was prevented from executing, the system would not be able to perform a “background code update” since all other threads would starve until the code update routines complete execution. The specification does not provide any description that would explain this apparent paradox.

7. Claims 2-11, 14-30, and 33-43 are rejected as being dependent upon a rejected base claim.

8. The following is a quotation of the second paragraph of 35 U.S.C. 112:

The specification shall conclude with one or more claims particularly pointing out and distinctly claiming the subject matter which the applicant regards as his invention.

Art Unit: 2192

9. Claims 1-44 are rejected under 35 U.S.C. 112, second paragraph, as being indefinite for failing to particularly point out and distinctly claim the subject matter which applicant regards as the invention.

10. Newly amended independent claims 1, 13, 31, 32, and 44 contain the phrase "...thereby prevent a task switching function..." The word "thereby" as used in the claims appears to make the associated steps optional. It is not clear if the use of the word and the associated limitations would limit the scope of the claim.

11. Claims 2-11, 14-30, and 33-43 are rejected as being dependent upon a rejected base claim.

*Claim Rejections - 35 USC § 103*

12. The text of the analysis of claims 2-44 is duplicated for convenience from the previous Office Action.

13. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

(a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negated by the manner in which the invention was made.

14. Claims 1, 4, 6, 8-11, 13, 14, 16, 18-21, 31, 32, 35, 37, and 39-42 are rejected under 35 U.S.C. 103(a) as being unpatentable over "Description of the Prior Art" (hereinafter "DPA") section appearing on pages 1-6 of the originally filed specification in view of U.S. Patent No. 6,023,704 to Gerard et al. (hereinafter "Gerard") in view of "Operating System Concepts" by Silberschatz and Galvin (hereinafter "Silberschatz").

In regard to claim 1, DPA discloses:

*A method (See page 2 line 23 – page 3 line 11) for performing a background code update of a current code image with an incoming code image in an embedded system, the method comprising the steps of:*

*(a) executing the current code image in the embedded system; See page 3 lines 5-6:*

Thus, the system firmware on the PROM may be updated because the current system firmware is executed from the RAM.

*(b) executing one or more code update routines from the incoming code image to update the current code image with the incoming code image; See page 3 lines 13-14:*

Utilizing code update routines from the firmware update itself has associated advantages.

*(c) executing a task switching function from the current code image to switch microprocessor control from executing the one or more code update routines of the incoming image to execute a function in the current code image. See page 3 lines 9-11:*

...a code update to the firmware may occur while the system operates normally utilizing one or more other process threads of the firmware, thereby accomplishing a background code update to the firmware.

In the context of “normally utilizing” threads with a background code update, this is clearly describing a process of “normally” switching tasks in order to provide the background update. Since task switching is normally handled by the “current code image,” normal utilization of threads in terms of task switching is handled by the current code image.

DPA does not expressly disclose: ...and thereby prevent a task switching function that originates from the one or more code update routines of the incoming image from

*executing*. However, in an analogous environment, Gerard teaches a method of updating a code image by modifying a method table pointer to point to the method table of another code image. See column 7 lines 27-28:

The method table pointer 410 is then changed to point to the methods of the new class (step 330).

According to this method, any method calls from a first image would then refer to a method body in the other image, thus preventing the execution of the method from the first image. Further, Silberschatz teaches an operating system performs scheduling, or task switching functions. See page 123:

Scheduling is a fundamental operating-system function.

Any code update routines that function separately from the operating system would use the operating system to schedule task switching.

It would have been obvious to one of ordinary skill in the art at the time the invention was made to use Gerard's method table pointers with DPA's task switching. One of ordinary skill would have been motivated to update the configuration of an existing code image (see Gerard column 4 lines 63-66). Further, it would have been obvious to one of ordinary skill in the art at the time the invention was made to use Silberschatz's scheduler with DPA's task switching routines. One of ordinary skill would have been motivated to schedule multiple processes in a multiprogramming environment (See Silberschatz, second paragraph in section 5.1 on page 123).

In regard to claim 4, the above rejection of claim 1 is incorporated. DPA further discloses loading the "current code image" into random access memory (RAM) for execution. See page 3 lines 2-4. DPA does not expressly disclose loading the incoming



Art Unit: 2192

code image into RAM. However, DPA further teaches that update routines from the incoming code image are executed. See page 3 lines 13-14. It would have been obvious to one of ordinary skill in the art at the time the invention was made to load the incoming image into RAM. One of ordinary skill would have been motivated to position the incoming code so as to allow the processor to access and execute the instructions found therein.

In regard to claim 6, the above rejection of claim 1 is incorporated. DPA further discloses: *providing a plurality of programmable memory devices for storing copies of the current code image; See page 3 lines 2-4. executing a copy of the current code image from one programmable memory device; See page 3 lines 2-4. updating a copy of the current code image in other programmable memory device with the incoming code image. See page 3 lines 4-6.*

In regard to claim 8, the above rejection of claim 1 is incorporated. DPA further discloses: *yielding microprocessor control by the executing function upon a task switching event; See page 2 lines 8-21, switching microprocessor control to continue executing the one or more code update routines to update the current code image with the incoming code image. See page 2 lines 27-30; also page 3 lines 8-11.*

In regard to claim 9, the above rejection of claim 8 is incorporated. DPA further discloses: *a step of continuing to switch microprocessor control between the one or more code update routines of the incoming code image and one or more functions of the current code image until the background code update completes. See page 3 lines 8-11.*

In regard to claim 10, the above rejection of claim 1 is incorporated. DPA further discloses: *wherein the task switching event is one selected from the group consisting of: round robin task switching; event driven task switching; and time slice task switching.* See page 2 lines 8-21.

In regard to claim 11, the above rejection of claim 8 is incorporated. DPA further discloses: *wherein the task switching event is one selected from the group consisting of: round robin task switching; event driven task switching; and time slice task switching.* All further limitations have been addressed in the above rejection of claim 10.

As per claim 13, DPA discloses: *An embedded system...* See page 1 line 19: “An embedded system...”. *A first programmable memory device...* See page 1 line 24: “PROM”, *A microprocessor* See page 1 line 22: “microprocessor”. All further limitations have been addressed in the above rejection of claim 1.

In regard to claim 14, the above rejection of claim 13 is incorporated. All further limitations have been addressed in the above rejection of claim 4.

In regard to claim 16, the above rejection of claim 13 is incorporated. DPA further discloses: *wherein the embedded system further comprises a second programmable memory device for storing a copy of the current code image, wherein the microprocessor executes the current code image from the first programmable memory device and updates the copy of the current code image in the second programmable memory device with the incoming code image.* See page 3 lines 1-6.

In regard to claims 18 and 19, the above rejection of claim 13 is incorporated. All further limitations have been addressed in the above rejections of claims 8 and 9, respectively.

In regard to claim 20, the above rejection of claim 13 is incorporated. All further limitations have been addressed in the above rejection of claim 10.

In regard to claim 21, the above rejection of claim 18 is incorporated. All further limitations have been addressed in the above rejection of claim 11.

In regard to claim 31, DPA discloses a storage automation library comprising an embedded system. See page 1 lines 24-27: "The larger system or machine that generally utilizes the embedded system may include a wide variety of systems, ranging from modems, to mid-range computing devices and enterprise systems, to **storage automation libraries**, to digital satellite receivers, and the like." All further limitations have been addressed in the above rejection of claim 13.

In regard to claim 32, DPA discloses a program storage device. See page 1 lines 21-24: "Typically, each of the embedded systems is housed on single microprocessor board with firmware (i.e., software) **stored as object code within a non-volatile memory device**, such as a programmable read only memory (i.e., 'PROM')." All further limitations have been addressed in the above rejection of claim 1.

In regard to claim 35, the above rejection of claim 32 is incorporated. All further limitations have been addressed in the above rejection of claim 4.

In regard to claims 37, and 39-42, the above rejection of claim 32 is incorporated. All further limitations have been addressed in the above rejections of claims 6, and 8-11, respectively.

15. Claims 2, 3, 28-30, 33, 34, and 44 are rejected under 35 U.S.C. 103(a) as being unpatentable over DPA, Gerard and Silberschatz as applied to claims 1, 4, 6, 8-11, 13, 14, 16, 18-21, 31, 32, 35, 37, and 39-42 above, and further in view of "Linkers and Loaders" by Levine (hereinafter "Levine").

In regard to claim 2, the above rejection of claim 1 is incorporated. DPA does not expressly disclose: wherein the method further comprises a step of retrieving an offset from the incoming code image for the one or more code update routines in the incoming code image. However, in an analogous environment, Levine teaches that the location of routines within a code segment can be represented by an offset from the beginning of the segment. See Levine section 7.4 starting on page 153. It would have been obvious to one of ordinary skill in the art at the time the invention was made to use Levine's teaching of relocation offsets with the routines of DPA. One of ordinary skill would have been motivated to use an offset that would allow access to software routines.

In regard to claim 3, the above rejection of claim 1 is incorporated. DPA does not expressly disclose: *wherein the method further comprises a step of retrieving an offset from the current code image of a task switching function.* However, DPA teaches relocation offsets as addressed in the above rejection of claim 2.

In regard to claim 28, the above rejection of claim 13 is incorporated. All further limitations have been addressed in the above rejections of claims 2 and 3.

In regard to claim 29, the above rejection of claim 28 is incorporated. Levine further discloses: *wherein the each respective code image comprises a header area for storing the offsets for the code update routines and the task switching function.* See pages 56, Figure 3.6. It would have been obvious to one of ordinary skill in the art at the time the invention was made to use Levine's header with DPA's code image. One of ordinary skill would have been motivated to store offsets in a header since this information is required before use of any particular code data or text, but is not actually part of the code.

In regard to claim 30, the above rejection of claim 13 is incorporated. DPA does not expressly disclose: *wherein offsets for the code update routines and the task switching function are stored at predetermined locations within each respective code image.* However, Levine teaches that relocation "fixups," or offsets, are stored at a predetermined location in a DOS EXE file. See Figure 3.6 on page 57. It would have been obvious to one of ordinary skill in the art at the time the invention was made to use Levine's teaching of predetermined offsets with DPA's routines. One of ordinary skill would have been motivated to find information regarding the location of a routine in a code image.

In regard to claims 33 and 34, the above rejection of claim 32 is incorporated. All further limitations have been addressed in the above rejections of claims 2 and 3, respectively.

In regard to claim 44, all further limitations have been addressed in the above rejections of claims 1, 2, and 3.

16. Claims 5, 15, 23, and 36 are rejected under 35 U.S.C. 103(a) as being unpatentable over DPA, Gerard and Silberschatz as applied to claims 1, 4, 6, 8-11, 13, 14, 16, 18-21, 31, 32, 35, 37, and 39-42 above, and further in view of "Understanding Computers: Input/Output" by Time-Life Books (hereinafter "Time-Life").

In regard to claim 5, the above rejection of claim 1 is incorporated. DPA does not expressly disclose: *wherein the method further comprises receiving the incoming code image into the embedded system via an input/output interface*. However, in an analogous environment, Time-Life teaches that computers gather and distribute digital information using an input/output (I/O) interface. See pages 26 and 27. It would have been obvious to one of ordinary skill in the art at the time the invention was made to use an I/O interface with the embedded system of DPA. One of ordinary skill would have been motivated to use hardware that is designed specifically to gather information when attempting to incorporate updated code from an external source.

In regard to claim 15, the above rejection of claim 14 is incorporated. All further limitations have been addressed in the above rejections of claim 5.

In regard to claim 23, the above rejection of claim 15 is incorporated. DPA does not expressly disclose: *wherein the embedded system comprises a bus for interconnecting one or more system components including the microprocessor, the random access*

*memory, the first programmable memory device, and the input/output interface.*

However, Time-Life teaches computer systems including a bus, microprocessor, RAM, programmable memory, and I/O interface. See the figure on pages 26 and 27. It would have been obvious to one of ordinary skill in the art at the time the invention was made to use Time-Life's teaching of computers with DPA's embedded system. One of ordinary skill would have been motivated to include common devices of computer system in an embedded system to provide a wide range of system functionality.

In regard to claim 36, the above rejection of claim 32 is incorporated. All further limitations have been addressed in the above rejection of claim 5.

17. Claims 7, 17, and 38 are rejected under 35 U.S.C. 103(a) as being unpatentable over DPA, Gerard, Silberschatz, and Levine as applied to claims 2, 3, 28-30, 33, 34, and 44 above, and further in view of U.S. Patent 4,974,191 to Amirghodsi et al. (hereinafter "Amirghodsi").

In regard to claim 7, the above rejection of claim 3 is incorporated. DPA and Levine do not expressly disclose: *wherein the method further comprises a step of testing the offset of the task switching function for validity before executing the task switching function.* However, in an analogous environment, Amirghodsi teaches testing a pointer for a NULL value to determine validity. See column 37 lines 53-59. It would have been obvious to one of ordinary skill in the art at the time the invention was made to use Amirghodsi's null test with DPA's offset. One of ordinary skill would have been

motivated to test the validity of the offset so that a null offset would not produce erroneous results.

In regard to claim 38, the above rejection of claim 34 is incorporated. All further limitations have been addressed in the above rejection of claim 7.

In regard to claim 17, the above rejection of claim 13 is incorporated. All further limitations have been addressed in the above rejection of claims 3 and 7.

18. Claim 24 is rejected under 35 U.S.C. 103(a) as being unpatentable over DPA, Gerard, Silberschatz, and Time-Life as applied to claim 23 above, and further in view of "Microsoft Computer Dictionary" published by Microsoft Press (hereinafter "MCD").

In regard to claim 24, the above rejection of claim 23 is incorporated. DPA discloses the integration of hardware and software onto a single microprocessor board. See page 1 lines 19-24. DPA and Time-Life do not expressly disclose: *wherein one or more of the system components form a part of an integrated microprocessor*. However, in an analogous environment, MCD further teaches that the process of integration combines multiple circuit elements on a single chip. See page 277 "integration". See also page 277, "integrated circuit". It would have been obvious to one of ordinary skill in the art at the time the invention was made to use MCD's teaching of integration to produce an integrated microprocessor. One of ordinary skill would have been motivated to integrate the various elements of a computing system in order to reduce production costs and package size of an embedded computing system.



19. Claims 12 and 43 are rejected under 35 U.S.C. 103(a) as being unpatentable over DPA, Gerard and Silberschatz as applied to claims 1, 4, 6, 8-11, 13, 14, 16, 18-21, 31, 32, 35, 37, and 39-42 above, and further in view of U.S. Patent 6237091 to Firooz et al. (hereinafter "Firooz").

In regard to claim 12, the above rejection of claim 1 is incorporated. DPA does not expressly disclose: *wherein the method further comprises a step of resetting the embedded system upon completion of the background code update*. However, in an analogous environment, Firooz teaches resetting a computer system upon completion of a code update. See Figure 2 and column 5 lines 10-15. It would have been obvious to one of ordinary skill in the art at the time the invention was made to use Firooz's teaching of resetting with DPA's background update. One of ordinary skill would have been motivated to reset a system after a code update in order to flush old code and execute the system using only new code.

In regard to claim 43, the above rejection of claim 32 is incorporated. All further limitations have been addressed in the above rejection of claim 12.

20. Claim 22 and 25-27 are rejected under 35 U.S.C. 103(a) as being unpatentable over DPA, Gerard and Silberschatz as applied to claims 1, 4, 6, 8-11, 13, 14, 16, 18-21, 31, 32, 35, 37, and 39-42 above, in view of Firooz and further in view of U.S. Patent 5132716 to Samuels et al. (hereinafter "Samuels").

In regard to claim 22, the above rejection of claim 13 is incorporated. DPA discloses executing a current code image and code update routines of an incoming code image as addressed above in the rejection of claim 13. DPA does not expressly disclose a bootloader or resetting the system. However, Firooz teaches resetting as addressed in the above rejection of claim 12. Also, in an analogous environment, Samuels teaches bootloaders for instructing a processor to execute a code image. See column 8 lines 25-49. It would have been obvious to one of ordinary skill in the art at the time the invention was made to use Firooz' resetting and Samuels' bootloader with DPA's code images. One of ordinary skill would have been motivated to provide a small initial code that initializes a system to prepare it for further execution of larger code.

In regard to claim 25, the above rejection of claim 22 is incorporated. DPA does not expressly disclose: *wherein the programmable memory device comprises a boot sector for storing the bootloader*. However, Samuels teaches boot sectors. See column 6 lines 39-48.

In regard to claim 26, the above rejection of claim 22 is incorporated. DPA further discloses: *wherein the bootloader tests the integrity of the current code image before instructing the microprocessor to execute it*. See column 7 lines 1-17.

In regard to claim 27, the above rejection of claim 22 is incorporated. DPA further discloses: *wherein the bootloader is enabled to check for availability of a code update and if the code update is available to initiate the code update*. See column 8 lines 25-29.

Art Unit: 2192

*Conclusion*

Any inquiry concerning this communication or earlier communications from the examiner should be directed to J. Derek Rutten whose telephone number is (571) 272-3703. The examiner can normally be reached on T-F 6:00 - 4:30.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Tuan Q. Dam can be reached on (571) 272-3695. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see <http://pair-direct.uspto.gov>. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

jdr



ANTONY NGUYEN-BA  
PRIMARY EXAMINER